

REMARKS

Claims 1, 2 and 13 have been amended. No claims have been added or canceled. Claims 1-22 remain pending in the application. Reconsideration is respectfully requested in light of the following remarks.

Section 103(a) Rejection:

The Examiner rejected claims 1-2, 5, 11-13, 16 and 22 under 35 U.S.C. § 103(a) as being unpatentable over Chamberlain et al. (U.S. Patent 6,434,744) (hereinafter “Chamberlain”) in view of Curtis (U.S. Patent 6,601,236) (hereinafter “Curtis”), claims 3, 4, 6, 7, 14, 15, 17 and 18 as being unpatentable over Chamberlain in view of Curtis and further in view of Moshir et al. (U.S. Publication 2002/0100036) (hereinafter “Moshir”), claims 8-10 and 19-21 as being unpatentable over Chamberlain in view of Curtis, Moshir and further in view of Taylor (U.S. Patent 6,161,218). Applicant respectfully traverses the rejections for at least the following reasons.

Claim 1

In regard to claim 1, the cited art fails to teach or suggest (i) deploying a patch package on a first computer running a first type of operating system, **wherein the patch package comprises a patching mechanism** and a first set of one or more new code components, and wherein the patching mechanism is also executable on a second computer running a second type of operating system, and (ii) executing the patching mechanism on the first computer, **wherein executing the patching mechanism comprises the patching mechanism performing a replacement of a first set of one or more old code components in a first application with the first set of one or more new code components**. In regard to “executing the patching mechanism...” the Examiner cites column 14, lines 30-36 of Chamberlain, which is reproduced below:

At step 834, the installer program 201 alters the program file by replacing the affected bits of the program file with the patch bits. This may be accomplished by the installer program 201 executing a patching routine

that replaces the affected bits with the new patch bits. Since techniques for applying patch bits to a program file to achieve an altered or updated program file are known to those skilled in the art, such techniques are not described here.

Presumably the Examiner considers “replacing the affected bits of the program file with the patch bits” (as taught by the cited art) to be equivalent to “performing a replacement of a first set of one or more old code components in a first application with the first set of one or more new code components” (as recited in Applicant’s claim). Furthermore, the cited art clearly teaches that the elements that perform “replacing the affected bits of the program file with the patch bits” are the “installer program 201” and “a patching routine” executed by the installer program. **However, in the system of the cited art, neither “installer program 201” nor the “patching routine” is part of the patch package, which directly contrasts with the specific limitations of Applicant’s claim.** More specifically, Applicant’s claim explicitly recites “wherein the patch package comprises a patching mechanism and a first set of one or more new code components” and “wherein executing the patching mechanism comprises the patching mechanism performing a replacement of a first set of one or more old code components in a first application with the first set of one or more new code components.” The cited art teaches a patch package (see e.g., Chamberlain, Fig. 5) that includes a summary portion 502, a transforms portion 504, and a cabinets portion 506. However, the patch package of the cited art does not include “installer program 201” nor does it include the “patching routine” of lines 30-36 of column 14 of Chamberlain. Since neither of the elements of the cited art on which the Examiner presumably relies to teach “performing a replacement of a first set of one or more old code components in a first application with the first set of one or more new code components” is a patching mechanism comprised within a particular patch package that is deployed on a computer system and also includes a first set of one or more new code components, the cited art cannot be said to teach or suggest the specific limitations of Applicant’s claim. The Curtis reference, which fails to mention anything about a patch package, fails to overcome the deficiencies of Chamberlain.

In the response to arguments section of the Final Office Action mailed February 2, 2009, the Examiner cites transforms 504 of Chamberlains patch package and asserts “[t]he transform information from the patch file provides the installer program with changes.” The Examiner’s remarks are not pertinent to the specific limitations of Applicant’s claim. Applicant’s claim does not recite a patch package that “provides the installer program with changes.” Instead, Applicant’s claim recites “wherein the patch package comprises a patching mechanism and a first set of one or more new code components” and “wherein executing the patching mechanism comprises the patching mechanism performing a replacement of a first set of one or more old code components in a first application with the first set of one or more new code components.” **No portion of the cited art teaches an element of a patch package “performing a replacement of a first set of one or more old code components in a first application with the first set of one or more new code components.” Instead, as described above, the cited art teaches that the elements that perform “replacing the affected bits of the program file with the patch bits” are “installer program 201” and “a patching routine” executed by the installer program. Furthermore, as also described above, the patch package of the cited art does not include “installer program 201” nor does it include the “patching routine” of lines 30-36 of column 14 of Chamberlain.**

Furthermore, even were the teachings of the cited art combined, the resulting combination would not teach the specific limitations of Applicant’s claim. At best, the proposed combination would result in a system where the installer program of Chamberlain (*see e.g.*, Chamberlain, Fig. 3, item 201) were modified according to the install program 17 of Curtis (*see e.g.*, Curtis, column 13, lines 29-47). However, such combination would only result in a modified installer program (which is not part of the patch package 301 of Chamberlain), not the deployment of a patch package that includes both a patching mechanism and a first set of one or more new code components. In fact, neither Chamberlain nor Curtis mention anything at all about such a patch package, much less deploying such a patch package. The Examiner fails to provide any response to this argument in the final Office Action mailed February 2, 2009.

Furthermore, the Examiner has not provided a proper reason as to why one of ordinary skill in the art would have combined the teachings of the cited references in a manner that would result in Applicants' claimed invention. The

Examiner asserts:

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to modify teaching of Chamberlain with the teachings of Curtis to include the patching mechanism is also executable on a second computer running a second type of operating system in order to provide a method using a cross-platform installation programs that is capable of installing and uninstallation a product on different operating systems and, at the same time, is capable of checking the storage space of the intended destination data storage drives of the operating system (col.3 lines 50-55) (emphasis added).

Modifying the teachings of Chamberlain “to include patching mechanism is also executable on a second computer running a second type of operating system” would not result in Applicant’s claimed invention. First, the Examiner has failed to explicitly state the element of the cited art that he considers to be equivalent to the claimed “patching mechanism.” It appears the Examiner is relying on installer program 201 or patching routine 340 to teach the patching mechanism of claim 1; however, neither of those elements is a patching mechanism comprised within a particular patch package that is deployed on a computer system and also includes a first set of one or more new code components. Accordingly, even were one to modify Chamberlain with the teachings of Curtis, the combination would not result in Applicant’s invention as claimed. As the Examiner is certainly aware, “[R]ejections on obviousness cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness” (emphasis added). *KSR v. Teleflex*, 550 U.S. ___, 82 USPQ2d 1385, 1396. **Since the Examiner has not provided such reasoning in order to explain how the combination of the cited references would result in Applicant’s invention according to the specific limitations of claim 1, the Examiner’s rejection is improper.**

Thus, for at least the reasons presented above, the rejection of claim 1 is unsupported by the cited art, and removal thereof is respectfully requested. Similar remarks apply to claim 12.

Applicant also asserts that numerous ones of the dependent claims recite further distinctions over the cited art. However, since the rejection has been shown to be unsupported for the independent claims, a further discussion of the dependent claims is not necessary at this time. Applicant reserves the right to present additional arguments.

CONCLUSION

Applicants submit the application is in condition for allowance, and notice to that effect is respectfully requested.

If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5681-54200/RCK.

Respectfully submitted,

/Robert C. Kowert/

Robert C. Kowert, Reg. #39,255
Attorney for Applicants

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: May 2, 2009